Name:	Section:	Fall 2010
ID:	CSC111 Final	Dr.

Question 1: (5 marks)

Give the output of the following program if the input from the keyboard is:

30 40 10 70 15 80 90

```
public class A {
  private int a[];
  private int val;
  private int nbValues;

  public A(int c, int v){
      a = new int[c];
      val = v;
      nbValues = 0;
  }

  public void insert(int v){
    if (nbValues<a.length && v>val){
        a[nbValues] = v;
        nbValues++;
    }
  }

  public int f(){
      int r = -1;
      if (nbValues>0) r = a[nbValues/2];
            return r;
  }

  public void display(){
    System.out.print("[");
    for (int i = 0;i<nbValues;i++)
            System.out.print(a[i]+" ");
    System.out.println("]");
  }
}
```

```
import java.util.Scanner;
public class Main1{
  public static void main(String a[]){
      Scanner s;
      A obj;
      int v;
      s = new Scanner(System.in);
      obj = new A(100,20);
      v = s.nextInt();
      while (obj.f()<50) {
  System.out.println("v="+v+"-->"+obj.f());
        obj.insert(v);
        v = s.nextInt();
      }
      obj.display();
  }
}
```

Name:	Section:	Fall 2010
ID:	CSC111 Final	Dr.

Question 2: (5 marks)

Find the errors in this java class

```
import java.util.Scanner;
public class myArray {

      public static void main(String[] args)  {

            Scanner read = new Scanner(System.in);
            int  age = new int[10];
            int  i = 0;
            Do {
                  System.out.print("Enter positive value of age[" + i + "]: ");
                  age[i] = read.next( );
                  i++;
            } while [ i < 10 ] ;

            Int M = 0;
            int j, index = -1;
            for  (j=0;  j++)  {
                  if (age[j] > M)  {
                        M = age( j );
                        index = j;

            }
            System.out.println("Greatest element of array age is: " + M);
            System.out.println(" and its position is: " + index);
      }
   }
```

Name:	Section:	Fall 2010
ID:	CSC111 Final	Dr.

Question 3: (4 Marks)

Convert the following Java class to its UML class representation: (4 marks)

```
public class Network {
  private String [] nodeIPs;   private String [] buffers;
  private int count;           private int server;
//----------------------
  public Network(int size) {
    nodeIPs = new String[size];    buffers = new String [size];
    count = 0;                     server = -1;
  }
//-----------------------
  private int searchNodeByIP(String ip) {
    for (int i=0; i<count; i++)
      if (nodeIPs[i].equals(ip)) return i;
    return -1;
  }
//------------------------
  public boolean addNode(String ip) {
    int i = searchNodeByIP(ip);
    if (i == -1 && count<nodeIPs.length) {
      nodeIPs[count] = ip;      buffers[count] = null;     count++;
      return true;
    }
    return false;
  }
//----------------------
  public boolean removeNode(String ip) {
    int i = searchNodeByIP(ip);
    if (i != -1) {
      nodeIPs[i] = nodeIPs[count-1];      buffers[i] = buffers[count-1];
      if (server == i) server = count-1;
      count--;
      return true;
    }
    return false;
  }
//----------------------
  public boolean transfer(String from, String to, String msg) {
    int f = searchNodeByIP(from);
    int t = searchNodeByIP(to);
    if (f != -1 && t != -1) {
      buffers[t] = from + "|" + msg;
      return true;
    }
    return false;
  }
//----------------------
  public void setServer(String ip) {
     int i = searchNodeByIP(ip);
     if (i != -1) server = i;
  }
//----------------------
  public String getServer() {
    if (server != -1) return nodeIPs[server];
    return "No server";
  }
}
```

Name:	Section:	Fall 2010
ID:	CSC111 Final	Dr.

Question 4:

Part A: (6 Marks) Having an AirTicket class (see the UML class diagram), we would like to manage its related information.

Airticket
-destination : string -business : boolean -actualPrice : double
+Airticket(in dest : string, in bsnss : boolean, in basicPrice : double) +getAmount() : double +getDestination() : string -calculateActualPrice(in basicPrice : double) : void +applyDiscount(in discount : double) : void

The AirTicket class has the following attributes:

- ***destination***: the destination of the flight (dammam, Tabouk, Jazan, etc.).
- ***business***: a boolean that indicates whether the ticket corresponds to a business class or not.
- ***actualPrice*:** the actual price of the ticket.

The class AirTicket provides the following services:

- A **constructor** accepts three parameters ***dest***, ***bsnss*** and ***basicPrice***. The two first ones are stored in the appropriate attributes and the third one is used to calculate the actual price by calling the method ***calculateAmount***.
- ***getAmount***: returns the amount.
- ***getDestination***: returns the destination.
- ***calculateActualPrice***: accepts one parameter ***basicPrice*** and calculates the air ticket's attribute ***actualPrice*** based on the type of the ticket (business or not). If the ticket is of type business there is an extra of 150 riyals.

  As an example suppose that the ticket basic price is 1300 and the ticket is a business class ticket then ***actualPrice*** = 1300+150 = 1450 riyals. If the price is 900 and the ticket is a non business class ticket then ***actualPrice*** = 900 riyals

- ***applyDiscount***: updates the attribute ***actualPrice*** based on the discount given as parameter if it is a positive number. As an example: suppose the amount is SR 1100 and the ticket has a discount of SR 250 then the ***actualPrice*** = 800 – 250 = 550 riyals (the amount should remain always positive).

Question : **implement the class AirTicket**

Part B: (8 Marks) Write a class ***Application*** that contains a *main* method, which performs the **following ordered processing**:

- read the number of air tickets nbTickets from the keyboards.
- Repeat the following nbTickets times: read the air ticket's data from the keyboard, create the object, and read the discount and apply .
- Display the average price and the destination of the ticket that has the maximum price.

Name:	Section:	Fall 2010
ID:	CSC111 Final	Dr.

Question 5: (12 Marks)

Having a UniversityStaff class (see the UML class diagram), we would like to manage the staff information.

UniversityStaff
-staffId : int[ ] -salary : double[ ] -collegeCode : int[ ] -nbCurrentStaff : int
+UniversityStaff(in size : int) +searchIndexOfStaffMember(in sId : int) : int +addStaffMember(in sId : int, in sal : double, in collgId : int) : void +displayStaffOfSpecificCollege(in collgId : int) : void +averageSalaryInSpecificCollege(in collgId : int) : double +lessStaff(in uni : UniversityStaff) : boolean

The UniversityStaff class has the following attributes:

- **staffId** : an array that stores the identifiers of the staff working in the university
- **salary**: an array that stores the salaries of the university staff.
- **collegeCode:** an array that stores the codes of the colleges of the university.
- **nbCurrentStaff:** the number of staff of the university.

The class University provides the following services:

- A **constructor** that accepts one parameter used to allocate the given *size* to the three arrays. The attribute *nbCurrentStaff* is initialized to zero.
- ***searchIndexOfStaffMember*** accepts one parameters *sId*. It searches the staff member that has this sId and returns the index (place location in the array) of that member if it is found, otherwise it returns -1.
- ***addStaffMember*** accepts three parameters *sId* as the staff identifier, *sal* as the salary and *collgId* as the code of the college in which that staff member is working and stores them in their appropriate arrays.
- ***displayStaffOfSpecificCollege*** accepts one parameter *collgId* as the code of a college and displays all staff id working in that college.
- ***averageSalaryOfSpecificCollege*** accepts one parameter *collgId* as the code of a college and returns the average salary of the staff members of that college.
- ***lessStaff*** accepts one parameter *univ* returns true if the current object has more staff than the object *univ* and returns false otherwise.

Question : Implement the class UniversityStaff